Machine Translation: An Integrated Approach

Kuang-hua Chen and Hsin-Hsi Chen

Department of Computer Science and Information Engineering

National Taiwan University

Taipei, Taiwan, R.O.C.

e-mail: khchen@nlg.csie.ntu.edu.tw, hh_chen@csie.ntu.edu.tw

Abstract

A pure statistics-based machine translation system is usually incapable of processing long sentences and is usually domain dependent. A pure rule-based machine translation system involves many costs in formulating rules. In addition, it is easy to introduce inconsistencies in a rule-based system, when the number of rules increases. Integrating both of approaches will get rid of these disadvantages. In this paper, a new model for machine translation system is proposed. A partial parsing method is adopted and the translation process is performed chunk by chunk. In synthesis module, the words are locally rearranged in chunks according to Markov model. Since the length of a chunk is much shorter than that of a sentence, the disadvantage of Markov model in dealing with long distance phenomena is greatly reduced. The structural transfer is fulfilled using a set of rules; in contrast, lexical transfer is resolved using bilingual constraints. The qualitative and quantitative knowledge is applied interleavingly and cooperatively, so that the advantages of both approaches are kept.

1. Introduction

Many different approaches (Bennett and Slocum, 1985; Brown *et al.*, 1990; Nagao, 1984; Mitamura *et al.*, 1991; Baker *et al.*, 1994) to machine translation system design have been proposed in literature. Traditional rule-based machine translation system (Bennett and Slocum, 1985) involves many human costs in formulating rules. That easily introduces inconsistencies, and it is too rigid to be robust. However, rules are usually universal, i.e., they are not domain dependent. In contrast, statistics-based machine translation system (Brown *et al.*, 1990) based on noise channel paradigm is robust in processing partial and ill-formed sentences. However, the computation time in a statistics-based system increases potentially with the length of sentences. In additional, the parameters strongly depend on the training corpus, i.e., it is domain dependent. The performance of an example-based system (Nagao, 1984) depends on the quality of collected examples and the similarity measure on examples and input sentences. When the matched units are subsentential structures (phrase structures), the performance of such a system is better than that of a word-level system. As for knowledge-based system (Mitamura *et al.*, 1991; Baker *et al.*, 1994), the difficulties are how to represent knowledge, how to build knowledge hierarchy, and how to infer knowledge. In addition, the cost of compiling knowledge is expensive.

A hybrid system is designed to integrate the advantages of these approaches and get rid of their disadvantages. We propose a transfer-based MT system augmented with probabilistic models in this paper. Both linguistic rules and parameterized knowledge are used in translation cooperatively. This paper is organized as follows. Section 2 gives an overview of

the proposed MT model. Sections 3, 4 and 5 discuss analysis module, transfer module and synthesis module respectively. Section 6 shows some experiments to demonstrate the feasibility of our proposed MT model. Section 7 provides some concluding remarks.

2. Overview of a New MT Model

A transfer-based machine translation system consists of analysis, transfer and synthesis modules. While receiving a n-word source sentence W_s, this MT system generates an l-word target sentence W_t. Let the intermediate forms for source part and target part be IF_s and IF_t, respectively. Equation 1 mathematically describes this translation model. This equation specifies that a form is dependent on all its pervious forms according to the traditional three-stage translation. However, the number of parameters contained in this model is inevitably too large to put it in practice. Some reduction should be carried out.

$$P(W_t|W_s) = \sum_{IF_s, IF_t} P(IF_s|W_s) \times P(IF_t|IF_s, W_s) \times P(W_t|IF_t, IF_s, W_s) \quad (1)$$

Assume that each form only depends on its proper previous form. Equation 1 is reformulated as equation 2.

$$P(W_t|W_s) \cong \sum_{IF_s, IF_t} P(IF_s|W_s) \times P(IF_t|IF_s) \times P(W_t|IF_t) \quad (2)$$

In actuality, the complexity of searching possible combination of these forms is also too high to endure. Further simplification in search should be made. The simplification is that the best form chosen in the previous stage is kept as proper one. We call such a simplified machine translation model a *pipelined* machine translation model. On the contrary, the original one is called a *conglomerated* machine translation model. The MT system proposed in this paper is based on the pipelined model and is shown in Figure 1.

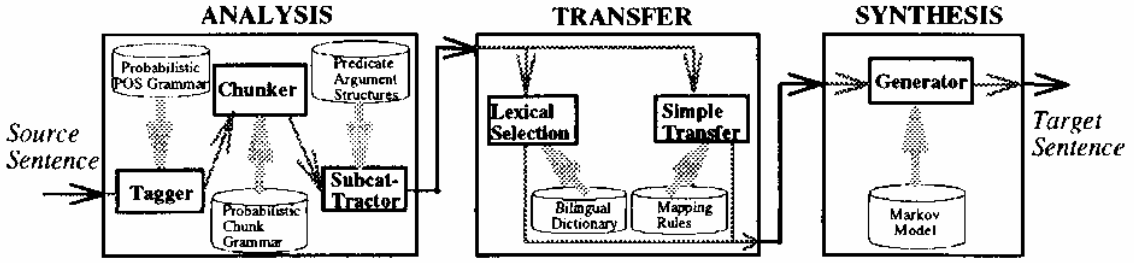

Figure 1. Our Translation Model

Analysis module is composed of a tagger, a chunker and a detector for predicate-argument structure. Transfer module consists of a lexical selection component and a simple transfer component. The knowledge needed in the module is the mapping rules of predicate-argument structures and a simple bilingual dictionary. The third module is the synthesis module. It consists of a generator and a bigram probability table.

3. Analysis Module

The analysis module is responsible for finding IF_s which has the largest $P(IF_s|W_s)$. In our model, the IF_s is composed of a chunk structure (C_s) and a predicate-argument structure (PA_s). That is,

$$P(IF_s|W_s) = P(C_s, PA_s|W_s) = P(C_s|W_s) \times P(PA_s|C_s, W_s)$$
$$= \sum_{T_s} (P(C_s|T_s, W_s) \times P(T_s|W_s)) \times P(PA_s|C_s, W_s)$$
$$\cong \sum_{T_s} (P(C_s|T_s) \times P(T_s|W_s)) \times P(PA_s|C_s, W_s) \quad (3)$$

where $T_s = ts_1, ts_2, ..., ts_n$, denotes the corresponding tags for $W_s = w_{s_1}, w_{s_2}, ..., w_{s_n}$. The $P(T_s|W_s)$ part is a probabilistic tagger; the $P(C_s|T_s)$ is a probabilistic chunker; the $P(PA_s|C_s,W_s)$ is implemented by 23 rules instead of a probabilistic model. Chunk structure is a linear chunk sequence. In contrast, predicate-argument structure provides dominated relation. Source sentence is first input to a probabilistic tagger, and then the corresponding sequence of tags is sent to the chunker. The tagger is trained by using LOB corpus (Johansson, 1986) and has more than 95% accuracy. The chunker (Chen and Chen, 1993), which determines the plausible boundaries of phrasal structures, segments the input tags into a chunk sequence. The best chunk sequence, $\hat{C}_s$, is found via equation 4.

$$\hat{C}_s = \underset{C_{s_i}}{\text{argmax}}\ P(C_{s_i}|T_s) = \underset{C_{s_i}}{\text{argmax}}\ P(C_{s_i}|ts_1^n) = \underset{C_{s_i}}{\text{argmax}}\ P_i(cs_1^{m_i}|ts_1^n)$$
$$\cong \underset{C_{s_i}}{\text{argmax}} \prod_{k=1}^{m_i} P(cs_k|cs_{k-1}, ts_1^n) \times P_i(cs_k|ts_1^n) \cong \underset{C_{s_i}}{\text{argmax}} \prod_{k=1}^{m_i} P_i(cs_k|cs_{k-1}) \times P_i(cs_k)$$
$$\cong \underset{C_{s_i}}{\text{argmax}} \sum_{k=1}^{m_i} [\log(P_i(cs_k|cs_{k-1})) + \log(P_i(cs_k))] \quad (4)$$

where $P_i(cs_1^{m_i}|ts_1^n)$ denotes the probability of the i'th chunk sequence and it contains m_i chunks. Note that an extra sentence initial marker denoted by c_0 is added. Dynamic programming technique shown in Algorithm 1 is used to find the best chunk sequence. The *score*[i] denotes the score for position i. The words between position *pre*[i] and position i form a best chunk from the viewpoint of position i. The *dscore*(c_i) is the score for the probability $P(c_i)$ and the *cscore*($c_i|c_{i-1}$) is the score for the probability $P(c_i|c_{i-1})$. These scores are trained by using a treebank, SUSANNE Corpus (Sampson, 1993; Sampson, 1995).

Algorithm 1: Chunker

Input: words $w_1, w_2, ..., w_n$, and the corresponding parts of speech $t_1, t_2, ..., t_n$

Output: a sequence of chunks $c_1, c_2, ..., c_m$

Method:

(1) score[0] = 0; pre[0] = 0;

(2) for (i = 1; i<n+1; i++) do 3 and 4;

(3) $j^* = \underset{0 \le j < i}{\text{argmax}}$ (score[pre[j]]+dscore(c_j)+cscore($c_j|c_{j-1}$));

where $c_j = t_{j+1}, ..., t_i$; $c_{j-1} = t_{pre[j]+1}, ..., t_j$;

(4) score[i]=score[pre[j^*]]+dscore(c_{j^*})+cscore($c_{j^*}|c_{j^*-1}$); pre[i] = j^*;

(5) for (i=n; i>0; i=pre[i]) do

output the word $w_{pre[i]+1}, ..., w_i$ to form a chunk;

The analysis model not only finds out a sequence of chunks, but also determines the predicate-argument structure. $P(PA_s|C_s,W_s)$ is used to describe the functionality of this part. Because it is easy to determine these structures using chunk sequence and word information, the current version of this component is rule-based. A finite state mechanism, Subcat-Tractor, is responsible for selecting one out of 23 predefined predicate-argument structures (Chen and Chen, 1994). The definition of these structures is a modified version of those in Oxford Advanced Learner's Dictionary (OALD) (Hornby, 1989).

4. Transfer Module

Transfer module consists of two components: lexical selection (lexical transfer) and simple transfer (structural transfer). That is,

$$\begin{aligned} P(IF_t|IF_s) &= P(C_t, PA_t|C_s, PA_s) \\ &= P(C_t|PA_t, C_s, PA_s) \times P(PA_t|C_s, PA_s) \\ &\cong P(C_t|C_s) \times P(PA_t|PA_s) \end{aligned} \quad (5)$$

IF_t is also composed of a chunk structure (C_t) and a predicate-argument structure (PA_t). Simple Transfer Mechanism maps the predicate-argument structure of source sentences into the counterpart of target sentences ($PA_s \rightarrow PA_t$). These predicate-argument structures are regarded as the skeleton of sentences. In other words, the simple transfer mechanism transfers source skeleton to target skeleton. The remaining "flesh" is adjusted in the synthesis module. So that overhead is reduced in the transfer stage. To transfer predicate-argument structures across languages, a set of rules is formulated. Each rule deals with one predicate-argument structure. These structures are determined in analysis module mentioned in Section 3. Some mappings from English predicate-argument structures to Chinese ones are direct. These predicate-argument structures are I, Ipr, Ip, La, Vn, Vt, Vnt, Vng, Vni, Tf, Tw, Tg, Tsg, Dnpr, Dnf, Dprf, Dnw, Dprw, and Dprt. The mapping rules for the rest predicate-argument structures are shown in Table 1.

Table 1. The Mapping Rules for Predicate-Argument Structures

	English Predicate-Argument Structure	Chinese Predicate-Argument Structure
Tnpr	arg0 verb arg1 preposition arg2	arg0 使 (把,將) arg1 verb arg2
Cna	arg0 verb arg1 adjective	arg0 使 (把,將) arg1 verb adjective
Cnn/a	arg0 verb arg1 as arg2	arg0 verb arg1 為 arg2
	arg0 verb arg1 as adjective	arg0 verb arg1 為 adjective
Vnn	arg0 verb arg1 arg2	arg0 使 (把,將) arg2 verb arg1

Transfer module is also responsible for lexical selection ($C_s \rightarrow C_t$). The lexical selection algorithm is presented on the basis of source word association norm and target word association norm (Church and Hanks, 1990). Word association norm of source and target languages can be trained independently from the corresponding corpora. Bilingual dictionary sets up the word correspondence. The proposed lexical selection algorithm chooses the most informative mates in source language as well as in target language. On the one hand, bilingual corpora are not needed in our approach, thus the difficulty in collecting large volume of bilingual texts for reliable statistics is avoided. On the other hand, the computation of this method is not complex. For example, the right Chinese counterpart of this sentence "flying

plane makes her duck" is "飛機使她迅速低頭". "Fly" has many senses in Chinese such as "飛", "逃出", *etc.* "Duck" has four readings in Chinese: "鴨子", "迅速低頭", "暫時沒入水中", and "水陸兩用車". Word pairs (make,duck), (her,duck), (make,her), (plane,her), (palne,make), and (fly,plane) have mutual information (*MI*) in descending order. The word pair (make,duck) has the highest *MI*, so both words are the most informative to each other. Then the senses of both words are determined by *MI* of target words. Therefore, the right senses, "使" and "迅速低頭", are selected and the senses of the two words are fixed. The second highest *MI* is word pair (her,duck). Since the sense of "duck" is fixed, the sense of "her" is the one has the highest *MI* with "迅速低頭". Using the same procedure, we could determine the sense of each word. After the procedure, the word of which sense is not fixed is assigned the most frequent sense. The detail algorithm is listed as follows.

Algorithm 2: Lexical Selection

Notation: S_i: the i'th word of the source sentence
T_{i_k}: the k'th target translation of the i'th word of the source sentence
$MI(E_i,E_j)$: mutual information of two expressions

Input: A sentence consists of n words, S_1, S_2, ..., S_n.

Output: A target word sequence.

Method: (1) Select the source word pair S_i and S_j that have the largest $MI(S_i,S_j)$, where at least one of their senses is not fixed.
(2) Select the target word pair T_{i_p} and T_{j_q} that have the largest $MI(T_{i_p},T_{j_q})$.
(3) Fix the target translation of S_i and S_j as T_{i_p} and T_{j_q}, respectively.
(4) Repeat (1) to (3) until every target translation of S_i (i = 1, 2, ..., n) is fixed.

5. Synthesis Module

The major task of synthesis module is sentence generation, so that this module is language-dependent. The final word order of target sentence is determined by global reordering and local reordering. The tasks of reordering are performed by simple transfer rule R discussed in Section 4. In contrast, the local reordering are carried out by synthesis module $P(W_t|C_t)$. The relation of global reordering and local reordering is shown in Figure 2.

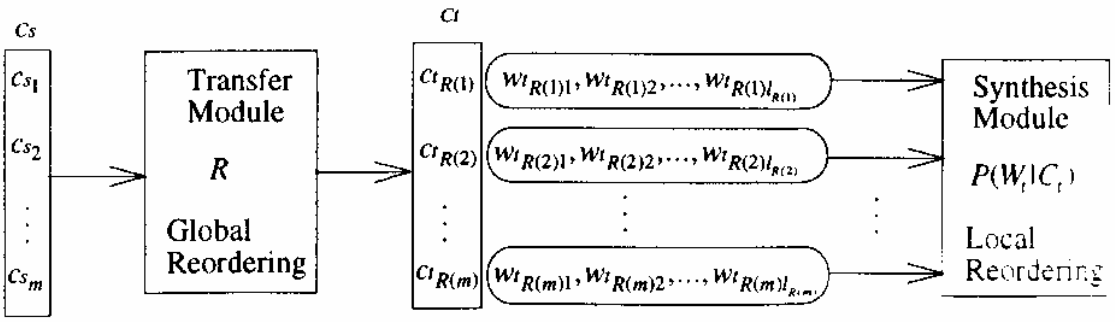

Figure 2. Global Reordering and Local Reordering

The model for synthesis module $P(W_t|IF_t)$ could be defined as:

$$P(W_t|IF_t) \approx P(W_t|C_t, PA_t) = P(W_t|C_t, R) = P(W_t|C_{t_{R(1)}}^{R(m)})$$
$$\cong \sum_{i=1}^{m} P(W_{t_{R(i)}}|C_{t_{R(i)}}) = \sum_{i=1}^{m} P(w_{t_{R(i)_1}}^{l_{R(i)}}|C_{t_{R(i)}})$$
$$\cong \sum_{i=1}^{m} P(w_{t_{R(i)1}}) \prod_{j=1}^{l_{R(i)}-1} P(w_{t_{R(i)j+1}}|w_{t_{R(i)j}}) \quad (6)$$

where the $l_{R(i)}$ denotes the number of words in target chunk $c_{t_{R(i)}}$. Corpus provides large volume of lively language phenomena. The implicit word order could be trained from corpus and could be expressed in probability. Therefore, many marginal phenomena could be covered naturally.

The target word order is captured by Markov model shown as equation 6. The disadvantage of Markov model is lack of capability to capture long distance phenomena. However, this disadvantage is reduced in our model, since long distance phenomena are less in chunks. In the study, an NTU Newspaper Corpus, a segmented Chinese corpus composed of texts from three major newspapers in Taiwan, is used as training text corpus. Table 2 lists the extracted statistic information.

Table 2. Statistic Information of the NTU Newspaper Corpus

Corpus	Total Words	Different Words	Word Bi-Gram
NTU Newspaper Corpus	2,636,793	43,262	921,633

6. Experiments and Evaluation

Total 200 sentences are used to test this proposed MT model. These testing sentences contain in transitive verb, transitive verbs, ditransitive verbs, prepositional phrases and some common constituents in English. Basically, testing sentences could be partitioned into three parts: NP + VP + PP. The PP might modify verbs in VP, nouns in VP or whole sentence. In other words, the testing suite stands for general phenomena in natural language.

The experimental results are evaluated by two factors. One is the word sense; the other is word order. For word sense, four grades are considered: *A*, *B*, *C* and *don't care*. By the subjective view of human, a target word is recognized as grade *A* (*C*) which depends on whether the meaning of the corresponding source word is expressed right (wrong) exactly. A target word is marked as *don't care*, if the corresponding source word is not found in the dictionary or it is an idiom which cannot be translated directly. Besides, the remaining are regarded as grade *B*. The score for word sense, *SWS*, is defined as

$$SWS = \frac{S_a \times \#\text{ of } A + S_b \times \#\text{ of } B + S_c \times \#\text{ of } C}{n - \#\text{ of don't care}} \quad (7)$$

$S_a = 1$, $S_b = 0.5$, $S_c = 0$ and n is number of words. For word order, the difference between exact word position (*EWP*) and generated word position (*GWP*) for each word is considered. Therefore, the difference of word order, *DWO*, is defined as

$$DWO = (\sum_{i=1}^{n} abs(EWP_i - GWP_i)) / n \quad (8)$$

Note that the high value of *SWS* means good results. On the contrary, the high value of *DWO* denotes bad results. These two factors together reflect the performance of the MT system. According to the evaluation method, the experimental results are shown in Table 3.

Table 3. Experimental Results

SWS	Number	*DWO*	Number
$0.9 \le SGS \le 1.0$	61	$0.0 \le DWO < 1.0$	143
$0.8 \le SGS < 0.9$	60	$1.0 \le DWO < 2.0$	28
$0.6 \le SGS < 0.8$	58	$2.0 \le DWO < 3.0$	16
$0.0 \le SGS < 0.6$	21	$3.0 \le DWO$	13

From Table 3, the performance of these testing sentences is promising. The number of sentences with *SWS* higher than 0.6 is 179 (89.5%) and the number of *DWO* values less than 1.0 dominates all distribution (71.5%). The efforts to recover the original word order from the generated word order are also considered. They are measured by the number of "key strokes" needed to recover original word order. Typically, to move a word needs a key stroke (the operations of cut and paste are seen as one key stroke). This measure is important for MT systems in post-editing phase. The distribution of key strokes is listed in Table 4. The average key strokes for recovering a sentence is 0.67.

Table 4. The Distribution of Key Strokes

Key Strokes	Number
0	116
1	50
2	19
3	15

7. Conclusion

In this paper, an integrated approach to machine translation design is proposed. Not only it has advantages of qualitative approach in processing core linguistic phenomena, but also keeps the advantages of quantitative approach in dealing with marginal linguistic phenomena. Since to fully understand sentences is not possible in near future, the proposed MT system does not completely parse input sentences. A partial parsing method is adopted and the translation process is performed chunk by chunk. In synthesis module, the word order is locally rearranged in chunks using Markov model. Since length of a chunk is much shorter than that of a sentence, the disadvantage of Markov model in dealing with long distance phenomena is greatly reduced. The structural transfer is fulfilled using a set of rules; in contrast, lexical transfer is resolved using mutual information which is trained from text corpora. The qualitative and quantitative knowledge is used interleavingly and cooperatively in the proposed

MT system. In summary, the integrated model is superior to pure noise channel model in the following ways:

- It consists of much finer modules than noisy channel model.
- The utilization of linguistic knowledge in this model is more natural than that in noisy channel model.

A testing suite containing general phenomena in language usage is used to evaluate the feasibility of the proposed MT system. The performance measures are based on word sense and word order. The experimental results show that the integrated approach to MT system have good performance in both measures. The post-editing efforts needed in this MT system are also few in the testing suite.

References

Baker, K. *et al.* (1994), "Coping with Ambiguity in a Large-Scale Machine Translation System," *Proceedings of COLING-94*, Kyoto, Japan, 1994, pp. 90-94.

Bennett, W. and Slocum, J. (1985), "The LRC Machine Translation System," *Computational Linguistics*, vol. 11, no. 2-3, 1985, pp. 111-119.

Brown, P. *et al.* (1990), "A Statistical Approach to Machine Translation," *Computational Linguistics*, vol. 16, no. 2, 1990, pp. 79-85.

Chen, K.H. and Chen, H.H. (1993), "A Probabilistic Chunker," *Proceedings of R.O.C. Computational Linguistics Conference VI*, 1993, pp. 99-117.

Chen, K.H. and Chen, H.H. (1994), "Acquiring Verb Subcategorization Frames," *Proceedings of the Second Conference for Natural Language Processing*, Vienna, Austria, September 28-30, 1994, pp. 407-410.

Church, K.W. and Hanks, P. (1990) "Word Association Norms, Mutual Information and Lexicography," *Computational Linguistics*, 1990, pp. 22-29.

Hornby, A.S. (1989), *Oxford Advanced Learner's Dictionary*, Oxford University Press, 1989.

Johansson, S. (1986), *The Tagged LOB Corpus: Users' Manual*, Bergen: Norwegian Computing Centre for the Humanities, 1986.

Mitamura, T.; Nyberg, E. and Carbonell, J. (1991), "An Efficient Interlingua Translation System for Multilingual Document Production," *Proceedings of Machine Translation Summit III*, Washington, DC, 1991, pp.

Nagao, M. (1984), "A Framework of Mechanical Translation between Japanese and English by Analogy Principle," *Artificial and Human Intelligence* (Elithorn, A. Eds.), 1984, pp. 173-180.

Sampson, G. (1993), "The SUSANNE Corpus," *ICAME Journal*, No. 17, 1993, pp. 125-127.

Sampson, G. (1995), *English for the Computer*, Oxford University Press, 1995.